Frequency Domain Filtering

Aubrey Olson
Matt Richard

Overview

- Project Overview
- Fourier Transformation Review
- Low pass filters
- High pass filters
- High frequency emphasis filters
- Homomorphic filters
- Timing fftw and qtimagelib FFT

Project Overview

- Fast Fourier Transform (FFT)
- We utilized both
    - FFTW (Fastest Fourier Transformation in the West)
    - The FFT contained in qtimagelib
- Filters Implemented
    - Ideal, Butterworth, Gaussian, Homomorphic

Why Frequency Domain Filtering?

DFT (FFT) is the single most important technique in image processing

FT Applications:

1. Digitization
2. Enhancement
3. Resortation
4. Encoding
5. Computer Tomography
6. Image Analysis

DFT and FFT

2D DFT: $O(N^4)$

$$F(u,v) = \Im\{f(x,y)\} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) e^{-\frac{i2\pi}{N}(ux+vy)}$$

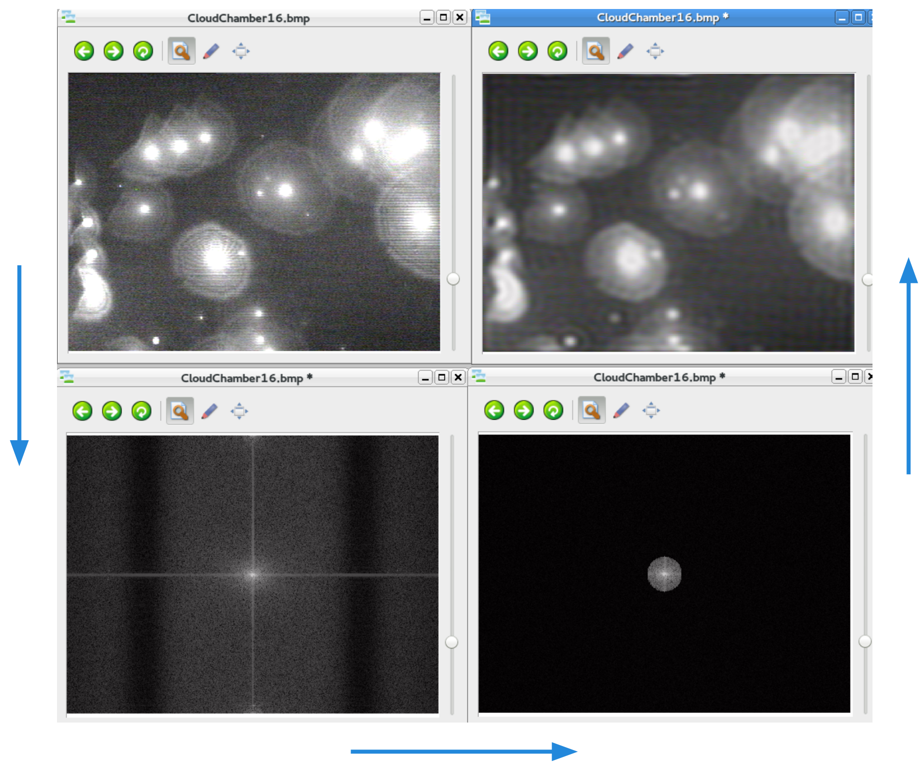
Separable: $O(N^3)$

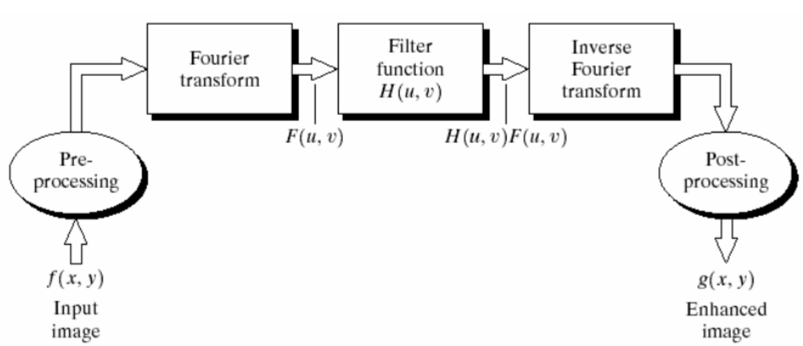
$$F(u,v) = \sum_{x=0}^{N-1} e^{-\frac{i2\pi ux}{N}} \sum_{y=0}^{N-1} f(x,y) e^{-\frac{i2\pi vy}{N}}$$

FFT: $O(N^2 \log N)$

Frequency Domain Filtering Review

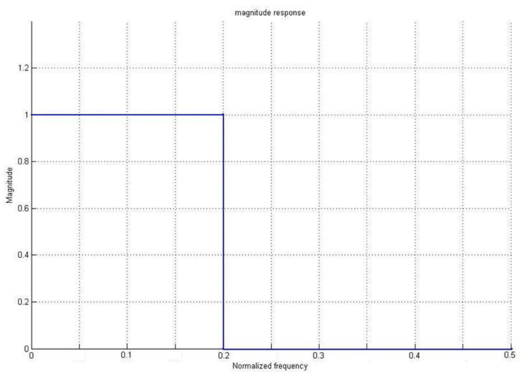
- Transform image to frequency domain
- Apply filter through multiplication
- Apply inverse FFT to return to spatial domain





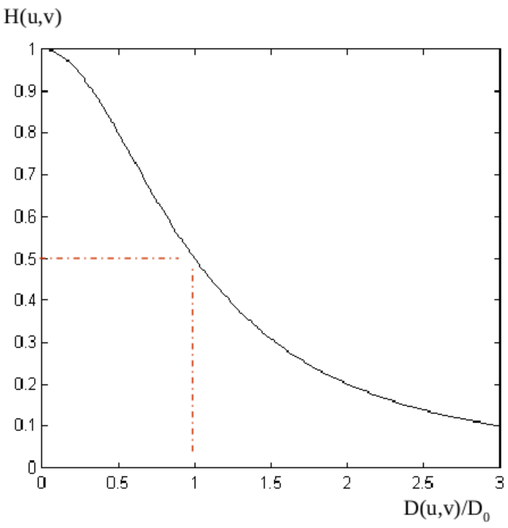
Low Pass Filters

Ideal



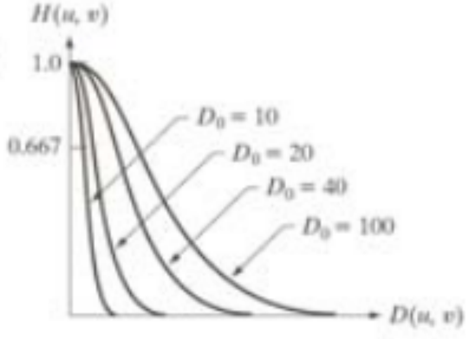
$$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$$

Butterworth



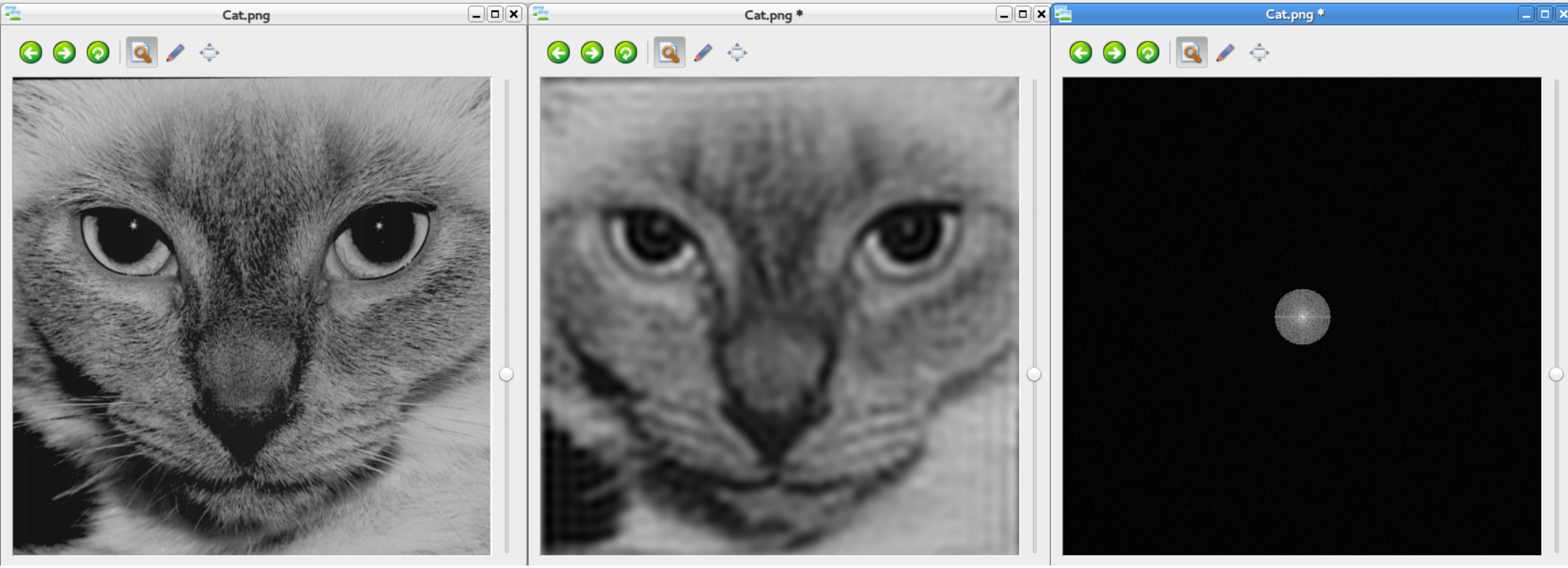
$$H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^{2n}}$$

Gaussian

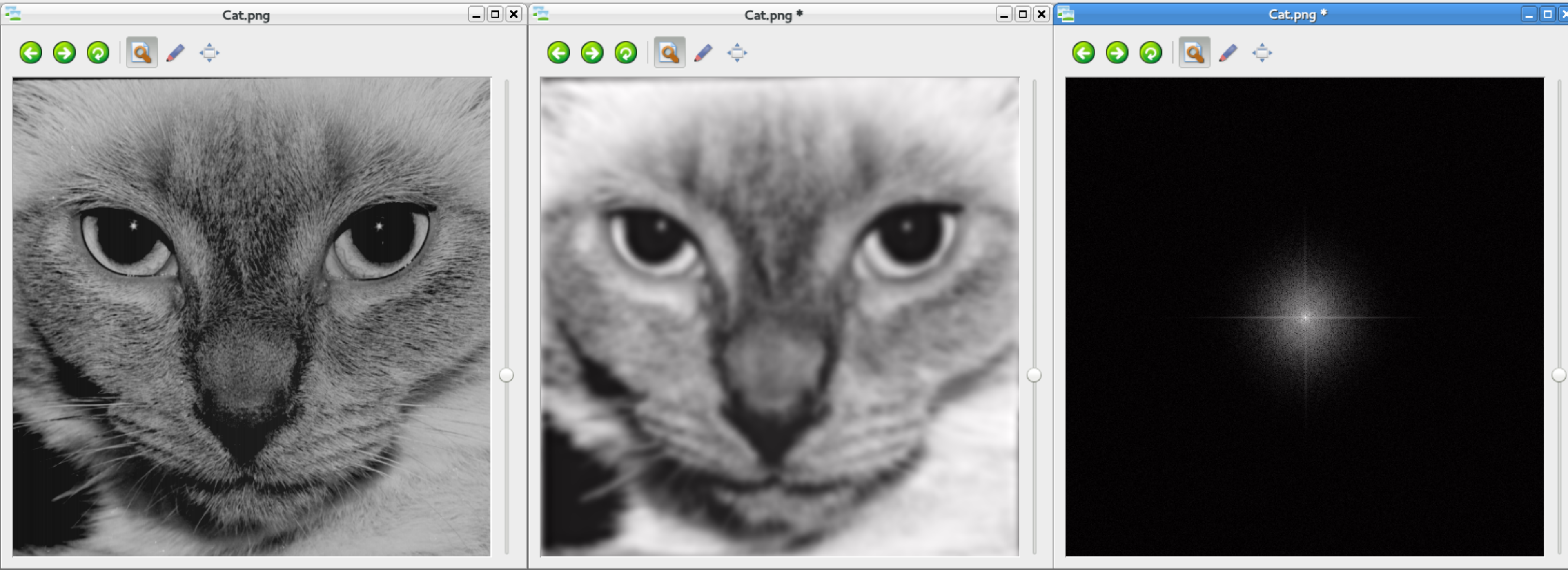


$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

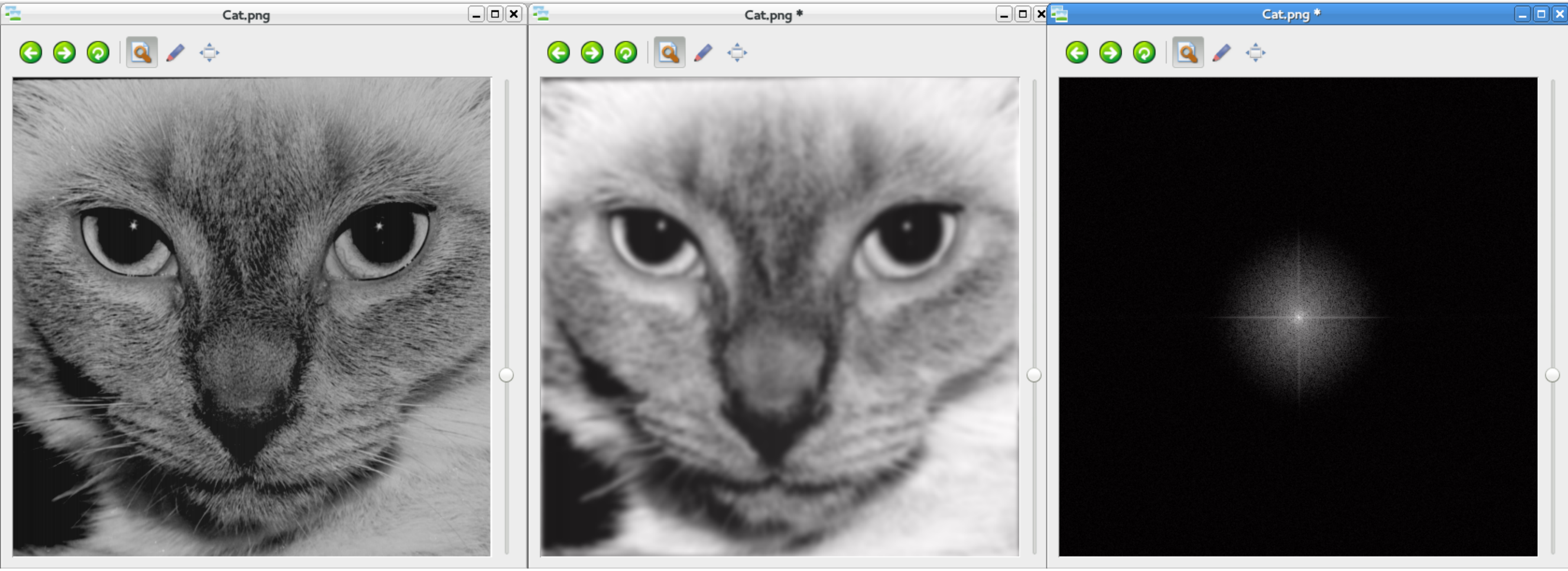
Ideal Low Pass



Butterworth Low Pass

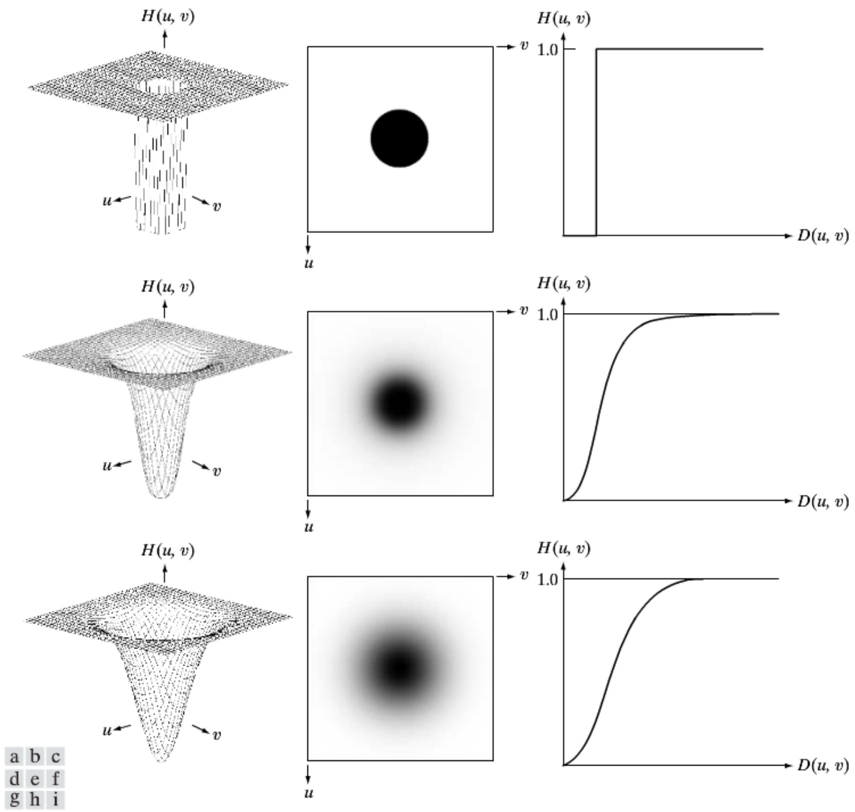


Gaussian Low Pass

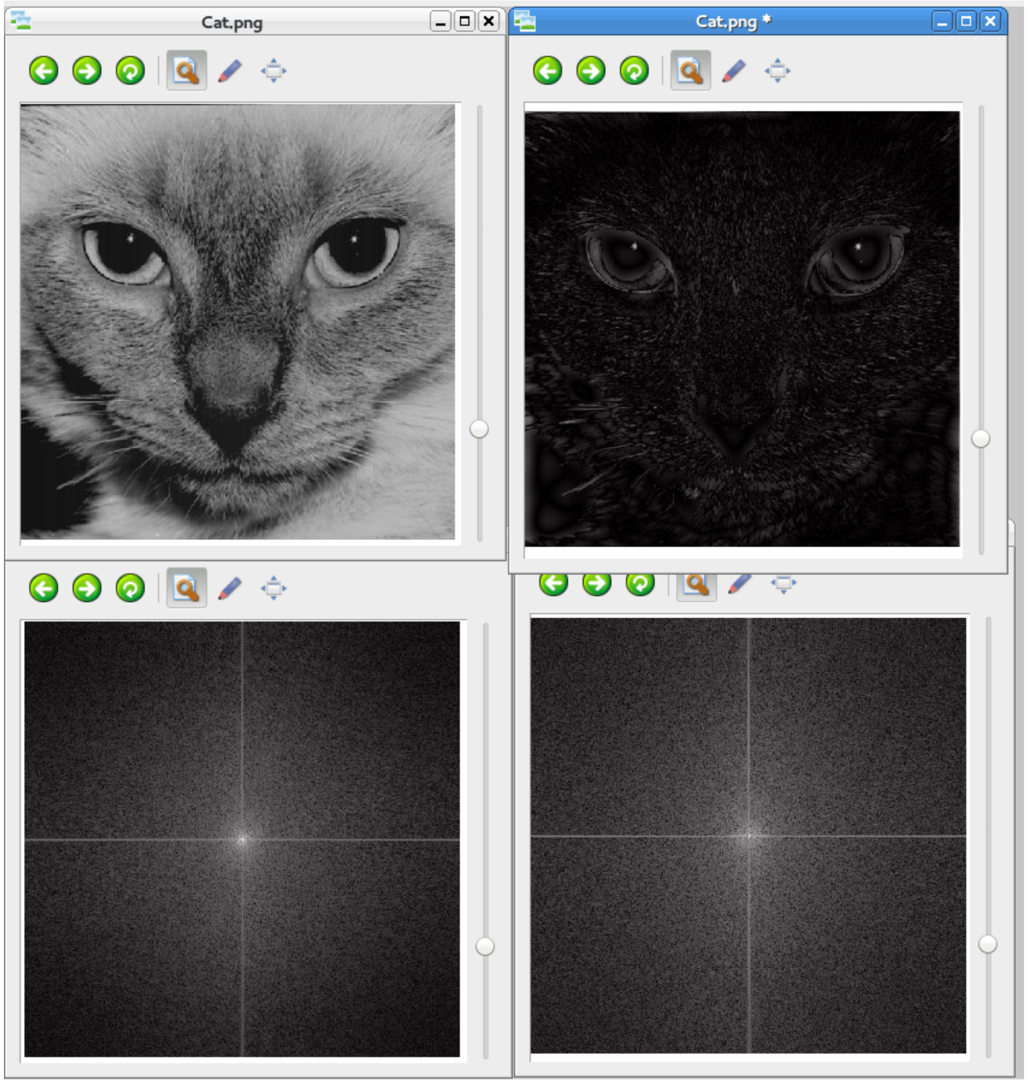


High Pass Filters

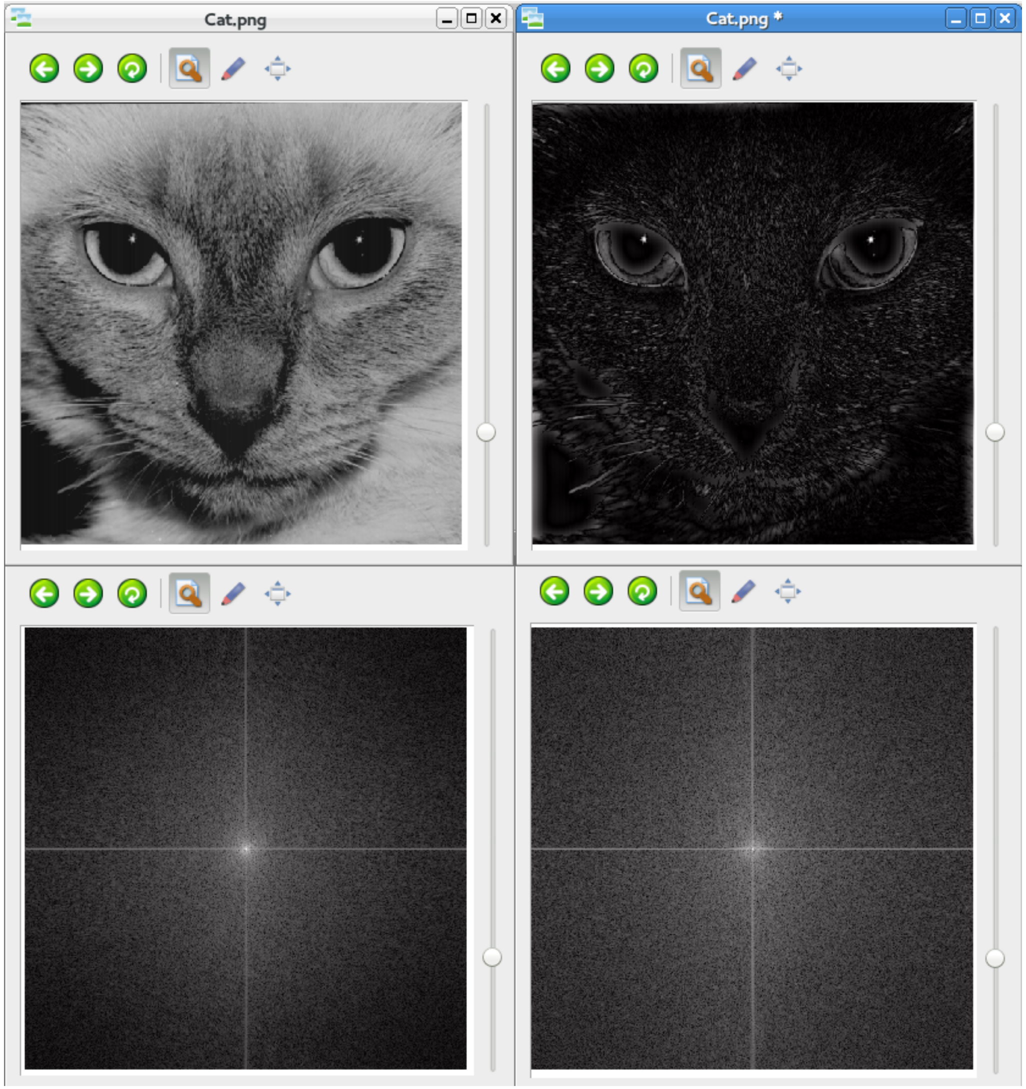
- Designed to restrict Frequency Domain to high frequencies
  - Ideal
  - Butterworth
  - Gaussian



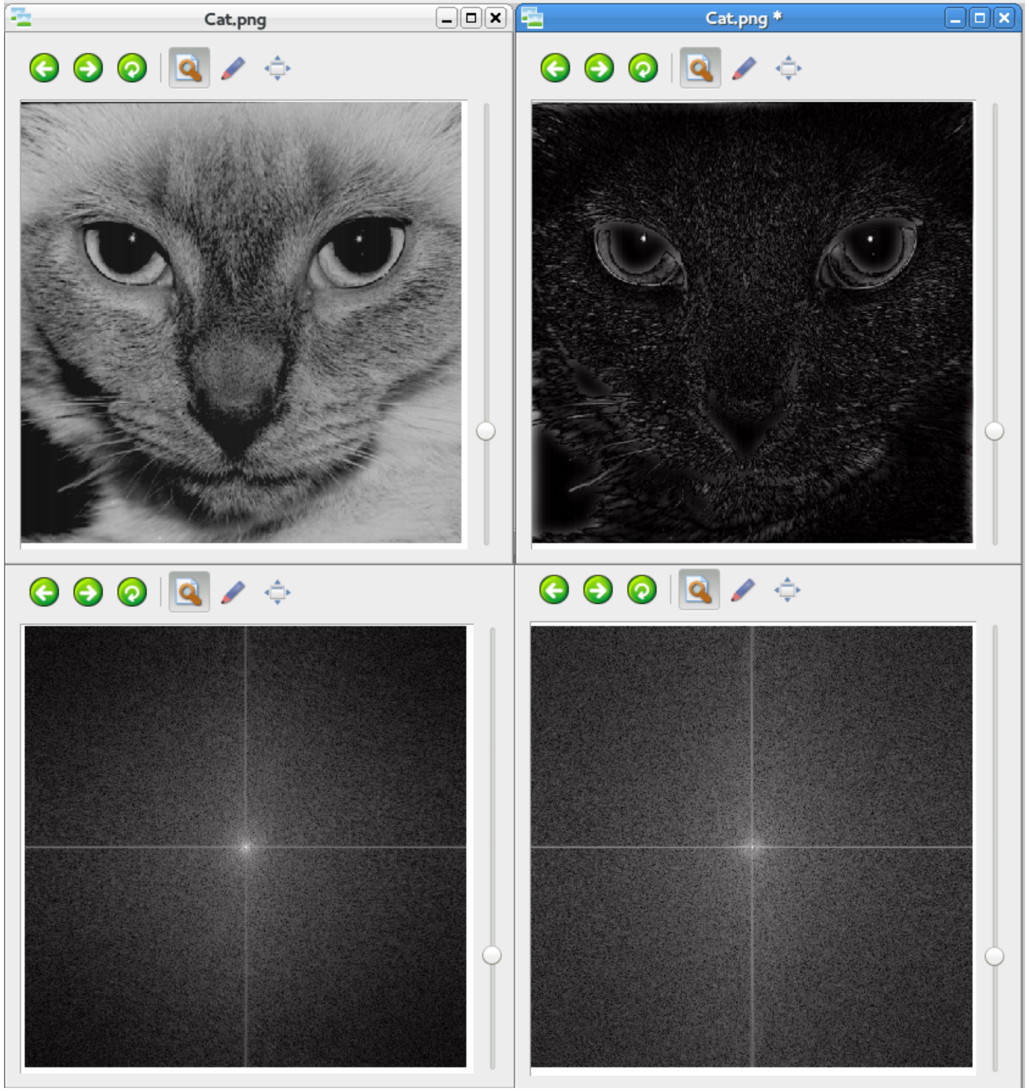
Ideal High Pass



Butterworth High Pass



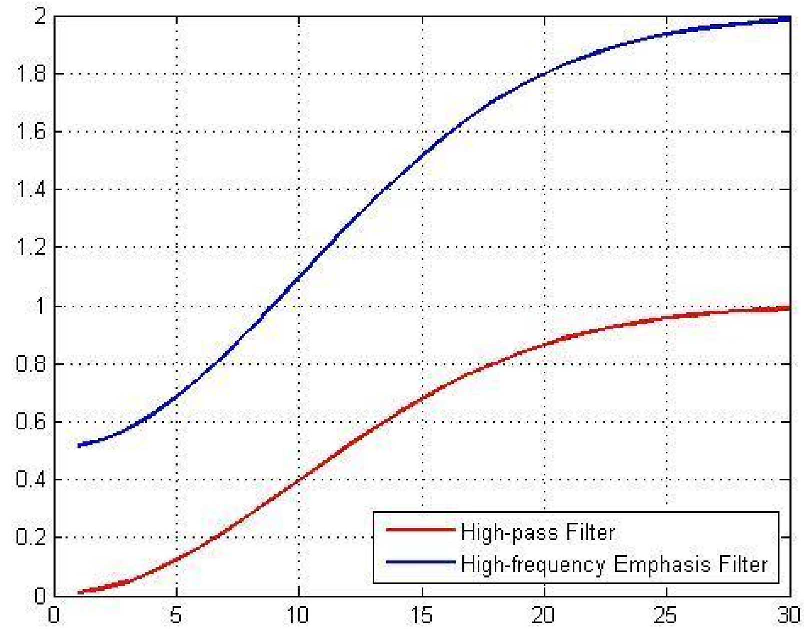
Gaussian High Pass



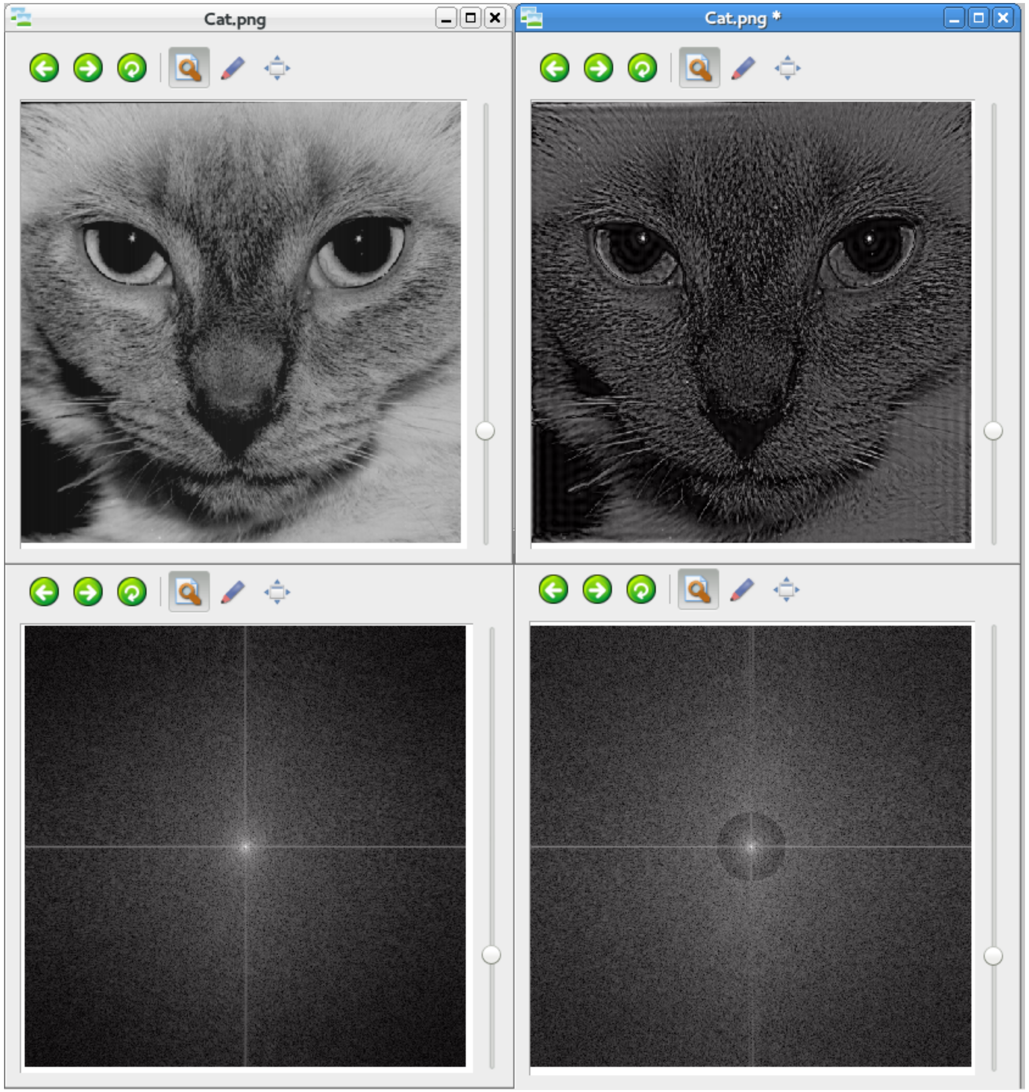
Emphasis Filters

- Designed to emphasize high frequencies
  - Ideal
  - Butterworth
  - Gaussian
  - Homomorphic

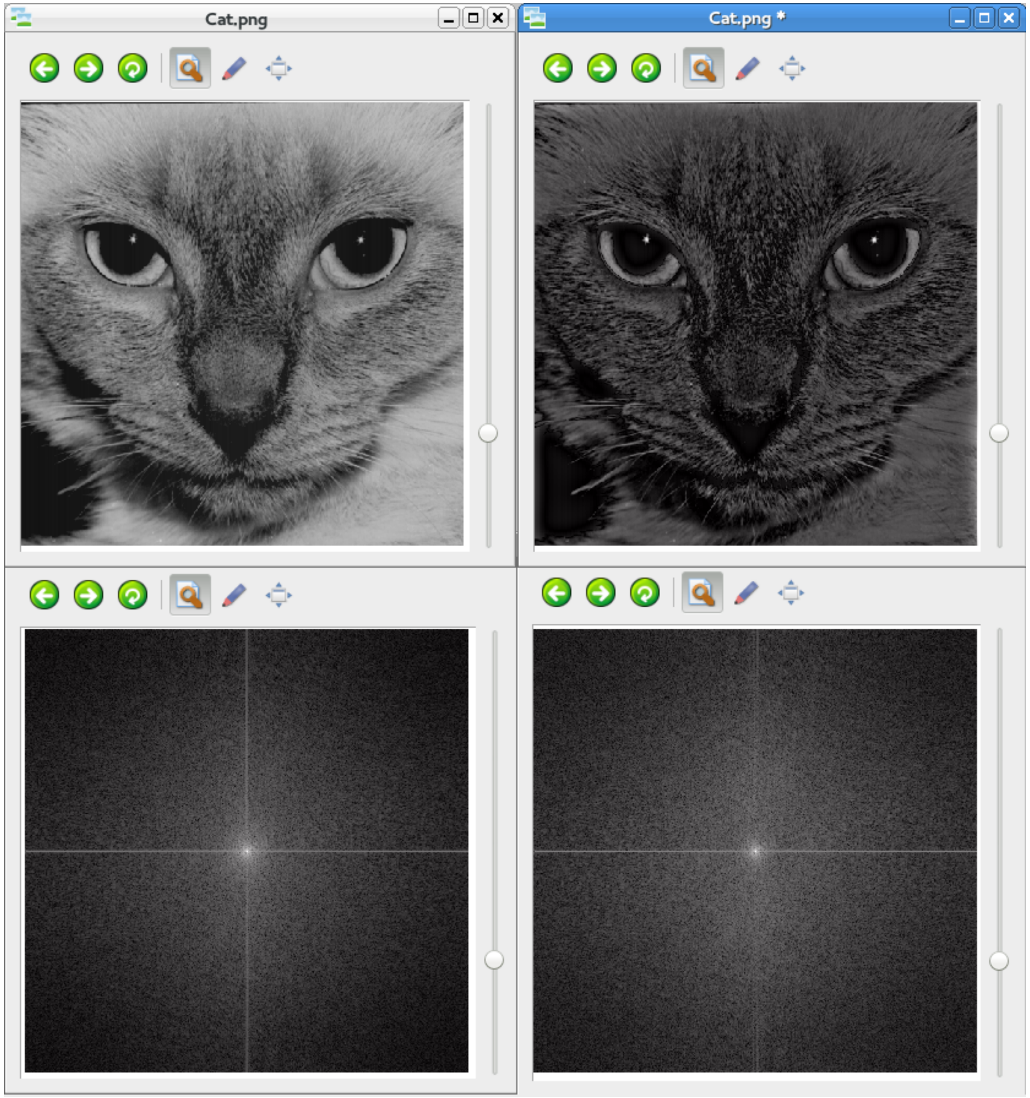
$$g(x,y) = \Im^{-1}\{[1 + k * H_{HP}(u,v)]F(u,v)\}$$



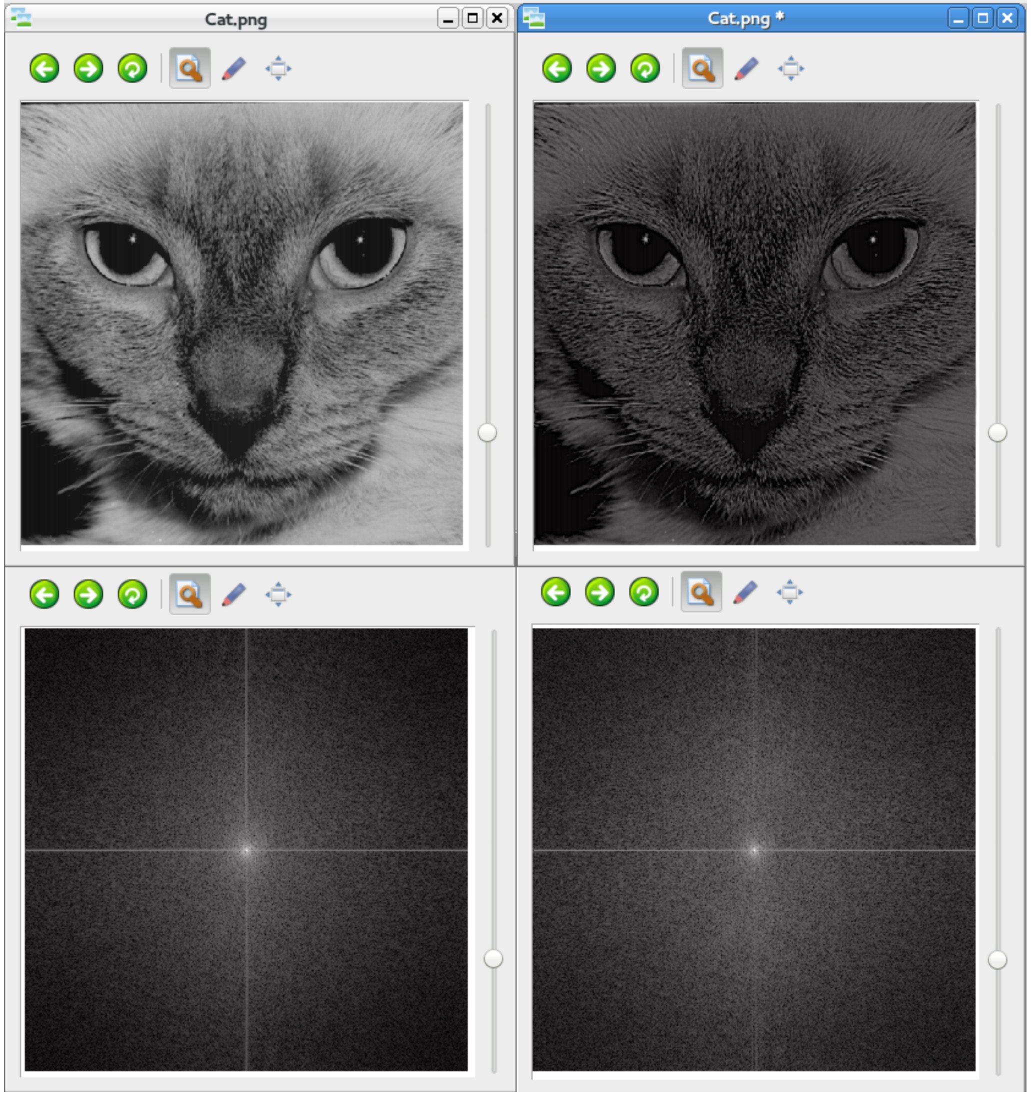
Ideal Emphasis Filter



Butterworth Emphasis Filter



Gaussian Emphasis



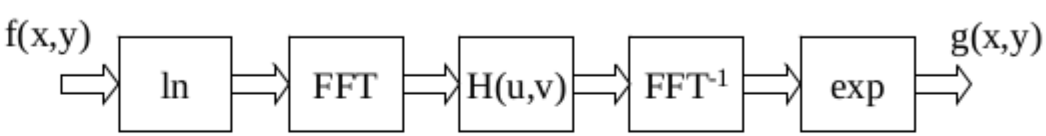
Homomorphic Filter

f(x,y)=i(x,y)r(x,y)

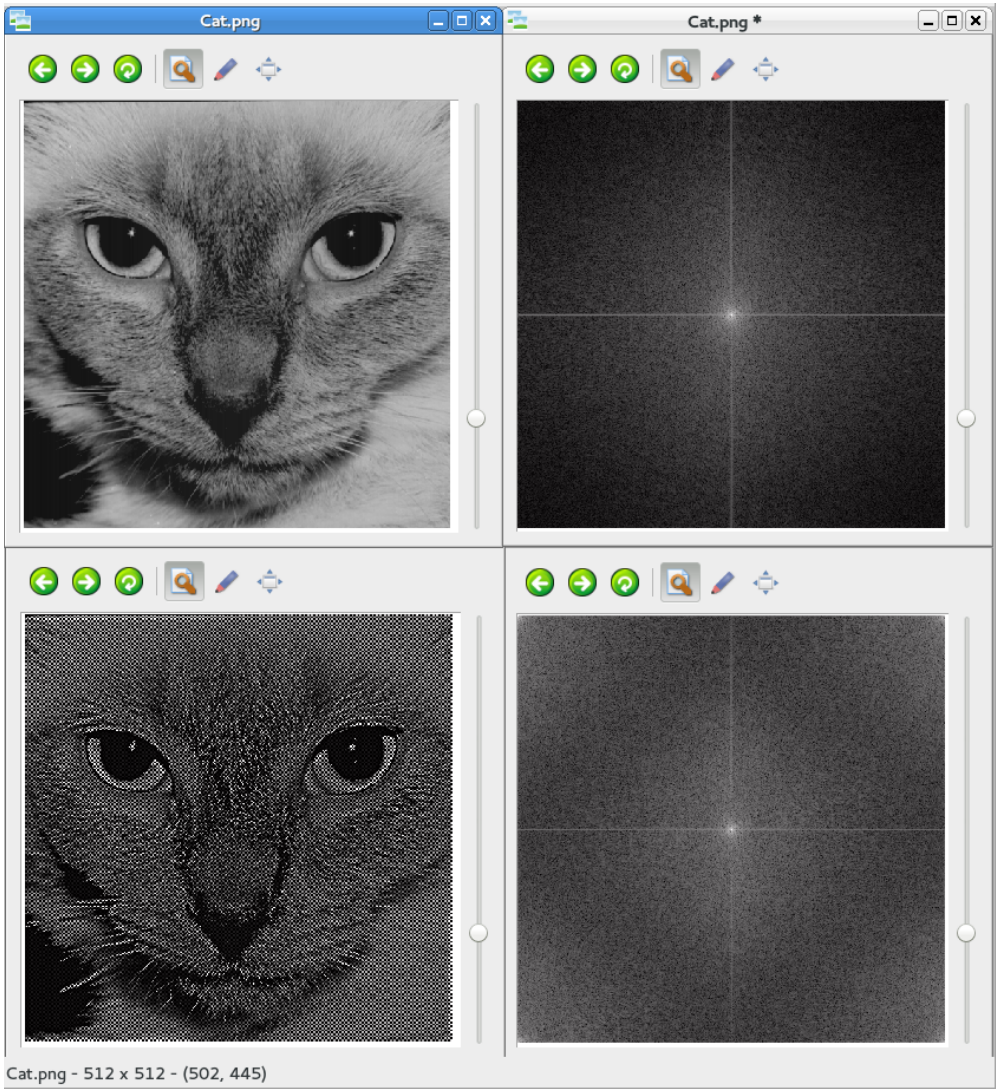
$$\Im\{f(x,y)\} \neq \Im\{i(x,y)\}\Im\{r(x,y)\}$$

$$\begin{aligned} z(x,y) &= \ln[f(x,y)] \\ &= \ln[i(x,y)] + \ln[r(x,y)] \end{aligned}$$

$$\begin{aligned} \Im\{z(x,y)\} &= \Im\{\ln[f(x,y)]\} \\ &= \Im\{\ln[i(x,y)]\} + \Im\{\ln[r(x,y)]\} \end{aligned}$$



Homomorphic Filter



2-D FFT (qtimagelib vs FFTW)

Cat Image (512x512):

	qtimagelib (s)	fftw (s)
Forward FFT	0.00202	0.00185
Inverse FFT	0.00185	0.00142

4K Image (4096x2304):

	qtimagelib (s)	fftw (s)
Forward FFT	0.134	0.0331
Inverse FFT	0.136	0.0317

Questions?

References

- http://jjackson.eng.ua.edu/courses/ece482/lectures/LECT09-4.pdf
- http://img210.imageshack.us/img210/8484/mpcsarbortfinal.jpg
- Digital Image Processing, Third Edition, Rafael C. Gonzales, Richard E. Woods. 2008
- http://www.mcs.sdsmt.edu/csc442/Resources/DIP_FFT.pdf